

[0016] Further, by setting the above pieces of information, for example, as the cells (pattern information) forming the mask pattern, an index can be created using, of the plurality of cells forming the mask pattern, those cells that have the same pattern arrangement. Furthermore, by setting the above pieces of data, for example, as the cells forming the mask pattern, an index can also be created by recognizing as one cell the region where the same cell repeatedly appears. That is, indexes can be applied to the mask pattern.

[0026] In the step of recognizing the cell repetition, a search is made for the largest one-dimensionally repeating same pattern (indicated by 0, \* indicates the cell being searched for) repeating in the x- and y-axis directions, as shown in Figure 3(1). Next, as shown in Figure 3(2), a search is made for the largest two-dimensionally repeating same pattern (indicated by 0, \* indicates the cell being searched for). Further, as shown in Figure 3(3), the largest same pattern repeating region is selected, and the search ends when there is no longer any repetition. The recognized repeating region is indicated by 0. The parameter for the recognized repeating region is stored, and the actual pattern corresponding to that parameter is erased from the hash table.



## PATENT ABSTRACTS OF JAPAN

(11) Publication number: **09128421 A**(43) Date of publication of application: **16.05.97**

(51) Int. Cl.

**G06F 17/50**  
**G03F 1/00**  
**G06F 5/00**  
**H01L 21/027**  
**H01L 21/82**

(21) Application number: **07281427**(22) Date of filing: **30.10.95**(71) Applicant: **SONY CORP**(72) Inventor: **ROBAATO FUERUTOMAN  
ASHIDA ISAO**

(54) **DATA PROCESSING METHOD AND DATA  
PROCESSING METHOD FOR MASK PATTERN**

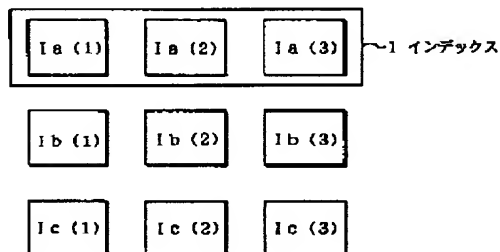
compress data.

COPYRIGHT: (C)1997,JPO

(57) Abstract:

**PROBLEM TO BE SOLVED:** To efficiently compress data and to simultaneously deal with non- geometrical information such as exposure information together with geometrical information such as shape information and size information.

**SOLUTION:** In this data processing method, many pieces of information are used. Among the many pieces of information la(1) to lc(3), plural specified information la(1) to la(3) are recognized as one piece of information to give index 1 and through the use of this index 1 and the other pieces of information lb(1) to lc(3), data is compressed and processed. In addition, when the many pieces of information la(1) to lc(3) consist of geometrical information and non- geometrical information, non-geometrical information is dealt with as geometrical information to simultaneously data-process both information. In addition, the area of a repeatedly appearing cell is recognized by repeated center tracking and furthermore the succeeding relation of constituting elements of the cell is obtained to



(19) 日本国特許庁 (J P)

## (12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-128421

(43) 公開日 平成9年(1997)5月16日

(51) Int.Cl. <sup>8</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 17/50			G 0 6 F 15/60	6 5 8 P
G 0 3 F 1/00			G 0 3 F 1/00	L
G 0 6 F 5/00			G 0 6 F 5/00	H
H 0 1 L 21/027			H 0 1 L 21/30	5 0 2 G
21/82			21/82	C
審査請求 未請求 請求項の数12 O L (全 14 頁)				

(21) 出願番号 特願平7-281427

(22) 出願日 平成7年(1995)10月30日

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72) 発明者 ロバート フェルトマン

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(72) 発明者 芦田 勲

東京都品川区北品川6丁目7番35号 ソニー株式会社内

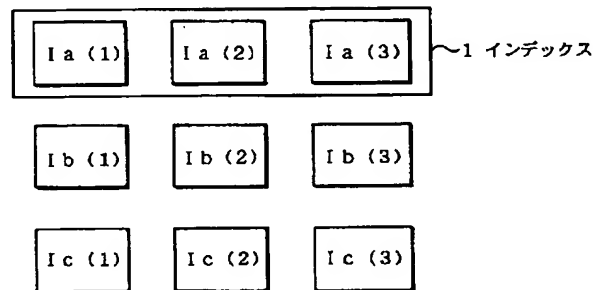
(74) 代理人 弁理士 船橋 國則

## (54) 【発明の名称】 データ処理方法およびマスクパターンのデータ処理方法

## (57) 【要約】

【課題】 マスクデータが巨大化しているため、効率的なデータの圧縮が必要であり、また露光情報のような非幾何学的情報を形状情報やサイズ情報のような幾何学的情報と同時に取り扱えるようにする必要があった。

【解決手段】 認識された多数の情報をを用いたデータ処理方法において、多数の情報 Ia(1) ~ Ic(3) のうちの特定の複数の情報 Ia(1) ~ Ia(3) を一つの情報として認識してインデックス 1 を与え、このインデックス 1 とともに残りの情報 Ib(1) ~ Ic(3) を用いて、データを圧縮して処理を行う方法である。また、多数の情報 Ia(1) ~ Ic(3) が、幾何学的情報と非幾何学的情報とからなる場合に、非幾何学的情報を仮想的な幾何学的情報として取り扱うことで、両者を同時にデータ処理する。また、繰り返しセンタトラッキングによって繰り返し表れるセルの領域を認識し、さらにセルの構成要素の継承関係を求めるとによってデータの圧縮を行う。



Ia (1) ~ Ia (3), Ib (1) ~ (3), Ic (1) ~ Ic (3) : 情報

本発明に係わる第1実施形態のインデックスの説明図

## 【特許請求の範囲】

【請求項1】 認識された多数の情報を用いたデータ処理方法において、

前記多数の情報のうちの特定の複数の情報を一つの情報として認識してインデックスを与え、該インデックスとともに残りの情報を用いて、データ処理することを特徴とするデータ処理方法。

【請求項2】 請求項1記載のデータ処理方法において、

前記多数の情報は、幾何学的情報と非幾何学的情報とからなり、

前記非幾何学的情報を仮想的な幾何学的情報として取り扱うことを特徴とするデータ処理方法。

【請求項3】 複数または単数のパターンからなるものでかつパターン配置が同一であるセルが繰り返し出現する領域を認識する処理を備えたデータ処理方法であって、

$x-y$ 座標系において、 $x$ 軸方向および $y$ 軸方向に繰り返す状態に配置された複数のセルのうちの一つのセルを該セルを含む $y$ 軸方向のセル群の中心に位置させることと、前記セルを該セルを含む $x$ 軸方向のセル群の中心に位置させることを順に繰り返して行うことで、前記セルを $x$ 軸方向および $y$ 軸方向に繰り返す状態に配置された複数のセル領域の中心に位置させることを特徴とするデータ処理方法。

【請求項4】 前記請求項3記載のデータ処理方法であって、

最大繰り返し数を $N$ として、繰り返し数 $n=N$ とし、まず $n>0$ か否かを判断する第1判断を行い、

前記第1判断の結果、 $n>0$ ならば、 $x$ 軸方向および $y$ 軸方向に連続的に配置された繰り返しセルのうちの一つのセルが、該セルを含む $y$ 軸方向のセル群の中心に位置していないか否かを判断する第2判断を行い、

前記第2判断の結果、前記セルが、該セルを含む $y$ 軸方向のセル群の中心に位置しない場合には、前記セルを前記 $y$ 軸方向のセル群の中心にずらし、かつ $n=n-1$ として、前記第2判断を繰り返し行い、

前記第2判断の結果、前記セルが、該セルを含む $y$ 軸方向のセル群の中心に位置している場合には、該セルを含む $x$ 軸方向のセル群の中心に位置していないか否かを判断する第3判断を行い、

前記第3判断の結果、前記セルが、該セルを含む $x$ 軸方向のセル群の中心に位置していない場合には、前記セルを前記 $x$ 軸方向のセル群の中心にずらし、かつ $n=n-1$ として、前記第2判断を繰り返し行い、

前記第3判断の結果、前記セルが、該セルを含む $x$ 軸方向のセル群の中心に位置している場合には、該セルの位置を $x$ 軸方向および $y$ 軸方向に連続的に配置された繰り返しセル領域の中心とし、そして終了し、

前記第1判断の結果、 $n\leq 0$ ならば終了することを特徴

とするデータ処理方法。

【請求項5】 認識された複数のセルからなるセル群を第1レベルとし、各セルに同じパターン集合が存在することを基本にして、前記第1レベルのセルからその一部のパターンを受け継いで前記第1レベルよりも上の第2レベルに新しいセルを構成するセル継承操作を、前記第1レベルのセルおよび新しく構成したセルに対して順次行って、前記第1レベルのセルからの継承関係を有するセルをツリー状に構成し、

続いて、各セルについてデータを圧縮するための効果を計算して、最もデータを圧縮するのに効果的なセルを選択して出力し、

その後、前記セル継承操作によって、残ったセルから継承関係を有する新しいセルをツリー状に再構成する操作を、前記第1レベルのセルが無くなるまであるいは前記セル継承操作によって継承するセルが見つからなくなるまで繰り返すことを特徴とするデータ処理方法。

【請求項6】 前記請求項5記載のデータ処理方法において、

前記セル継承操作は、指定されたlevelにあるデータ処理方法TREE[level]を実行するに際して、 $a(level)=1$ とし、指定されたlevelに登録されているセル数Nlibs[level]が $a(level)\leq Nlibs[level]$ か否かを判断する第1判断を行い、

前記第1判断の結果、 $a(level)\leq Nlibs[level]$ の場合には、指定されたlevelの $a(level)$ 番目に登録されているセルlibs[level][ $a(level)$ ]をセルAとし、続いて $b(level)=a(level)+1$ とし、

続いて、指定されたlevelに登録されているセル数Nlibs[level]が、 $b(level)\leq Nlibs[level]$ か否かを判断する第2判断を行い、

前記第2判断の結果、 $b(level)>Nlibs[level]$ の場合には、 $a(level)=a(level)+1$ とし、前記第1判断を繰り返し行い、

前記第2判断の結果、 $b(level)\leq Nlibs[level]$ の場合には、指定されたlevelの $b(level)$ 番目に登録されているセルlibs[level][ $b(level)$ ]をセルBとし、

続いて前記セルAと前記セルBとに共通するセルCを構成し、該セルCが空であるか否かを判断する第3判断を行い、

前記第3判断の結果、セルCが空である場合には、 $b(level)=b(level)+1$ として、前記第2判断を繰り返し行い、

前記第3判断の結果、セルCが空でない場合には、すでに登録されているセルにセルCと同一パターンのセットを持つセルが存在するか否かを判断する第4判断を行い、

3

前記第 4 判断の結果、セル C と同一パターンのセットを持つセルが存在する場合には、 $b(\text{level}) = b(\text{level}) + 1$  として、前記第 2 判断を繰り返し行い、  
 前記第 4 判断の結果、セル C と同一パターンのセットを持つセルが存在しない場合には、当該レベルよりも一つ上のレベルに  $N\text{libs}[\text{level} + 1] + 1$  番目のセルを増設し、 $N\text{libs}[\text{level} + 1] = N\text{libs}[\text{level} + 1] + 1$  とし、該増設したセル  $\text{libs}[\text{level} + 1]$  ( $N\text{libs}[\text{level} + 1]$ ) に前記セル C を割り当てて登録し、  
 続いて  $\text{level} = \text{level} + 1$  として、一つ上の  $\text{level}$  に対して  $\text{TREE}[\text{level}]$  を再帰的に実行するため、前記  $a(\text{level}) = 1$  に戻り、  
 前記第 1 判断の結果、 $a(\text{level}) > N\text{libs}[\text{level}]$  の場合には、 $\text{level} \leq 1$  か否かを判断する第 5 判断を行い、  
 前記第 5 判断の結果、 $\text{level} > 1$  の場合には、 $\text{level} = \text{level} - 1$  として、一つ前の  $\text{level}$  の  $\text{TREE}[\text{level}]$  に戻り、 $b(\text{level}) = b(\text{level}) + 1$  として、  
 前記第 2 判断を繰り返し行い、  
 前記第 5 判断の結果、 $\text{level} \leq 1$  の場合には、終了することを特徴とするデータ処理方法。

【請求項 7】 認識された多数のパターン情報を用いたマスクパターンのデータ処理方法において、  
 前記多数のパターン情報のうちの特定の複数のパターン情報を一つのパターン情報として認識してインデックスを与え、該インデックスとともに残りのパターン情報を用いて、データ処理することを特徴とするマスクパターンのデータ処理方法。

【請求項 8】 請求項 7 記載のマスクパターンのデータ処理方法において、  
 前記多数のパターン情報は、幾何学的情報と非幾何学的情報とからなり、  
 前記非幾何学的情報を仮想的な幾何学的情報として取り扱うことを特徴とするマスクパターンのデータ処理方法。

【請求項 9】 複数または単数のパターンからなるものでかつ露光装置に入力されるマスクパターン配置が同一であるセルが繰り返し出現する領域を認識する処理を備えたマスクパターンのデータ処理方法であって、  
 $x-y$  座標系において、 $x$  軸方向および  $y$  軸方向に繰り返す状態に配置された複数のセルのうちの一つのセルを該セルを含む  $y$  軸方向のセル群の中心に位置させると、前記セルを該セルを含む  $x$  軸方向のセル群の中心に位置させることを順に繰り返して行うことで、  
 前記セルを  $x$  軸方向および  $y$  軸方向に繰り返す状態に配置された複数のセル領域の中心に位置させることを特徴とするマスクパターンのデータ処理方法。

【請求項 10】 請求項 9 記載のマスクパターンのデータ処理方法であって、  
 最大繰り返し数を  $N$  として、繰り返し数  $n = N$  とし、

4

まず  $n > 0$  か否かを判断する第 1 判断を行い、

前記第 1 判断の結果、 $n > 0$  ならば、 $x$  軸方向および  $y$  軸方向に連続的に配置されたマスクパターンを構成する繰り返しセルのうちの一つのセルが、該セルを含む  $y$  軸方向のセル群の中心に位置するか否かを判断する第 2 判断を行い、

前記第 2 判断の結果、前記セルが、該セルを含む  $y$  軸方向のセル群の中心に位置しない場合には、前記セルを前記  $y$  軸方向のセル群の中心にずらし、かつ  $n = n - 1$  として、前記第 2 判断を繰り返し行い、

前記第 2 判断の結果、前記セルが、該セルを含む  $y$  軸方向のセル群の中心に位置している場合には、該セルを含む  $x$  軸方向のセル群の中心に位置していないか否かを判断する第 3 判断を行い、

前記第 3 判断の結果、前記セルが、該セルを含む  $x$  軸方向のセル群の中心に位置していない場合には、前記セルを前記  $x$  軸方向のセル群の中心にずらし、かつ  $n = n - 1$  として、前記第 2 判断を繰り返し行い、

前記第 3 判断の結果、前記セルが、該セルを含む  $x$  軸方向のセル群の中心に位置している場合には、該セルの位置を  $x$  軸方向および  $y$  軸方向に連続的に配置された繰り返しセルの中心とし、そして終了し、

前記第 1 判断の結果、 $n \leq 0$  ならば終了することを特徴とするマスクパターンのデータ処理方法。

【請求項 11】 認識されたものでかつ露光装置に入力されるマスクパターンを構成する複数のセルからなるセル群を第 1 レベルとし、各セルに同じパターン集合が存在することを基本にして、前記第 1 レベルのセルからその一部のパターンを受け継いで前記第 1 レベルよりも上の第 2 レベルに新しいセルを構成するセル継承操作を、  
 前記第 1 レベルのセルおよび新しく構成したセルに対して順次行って、前記第 1 レベルのセルからの継承関係を有するセルをツリー状に構成し、  
 続いて、各セルについてデータを圧縮するための効果を計算して、最もデータを圧縮するのに効果的なセルを選択して出力し、

その後、前記セル継承操作によって、残ったセルから継承関係を有する新しいセルをツリー状に再構成する操作を、前記第 1 レベルのセルが無くなるまであるいは前記セル継承操作によって継承するセルが見つからなくなるまで繰り返すことを特徴とするマスクパターンのデータ処理方法。

【請求項 12】 請求項 11 記載のマスクパターンのデータ処理方法であって、  
 前記セル継承操作は、  
 指定された  $\text{level}$  にあるデータ処理方法  $\text{TREE}[\text{level}]$  を実行するに際して、 $a(\text{level}) = 1$  とし、  
 指定された  $\text{level}$  に登録されているセル数  $N\text{libs}[\text{level}]$  が  $a(\text{level}) \leq N\text{libs}[\text{level}]$  か否かを判断する第 1 判断を行い、

5

前記第1判断の結果、 $a(\text{level}) \leq N\text{libs}[\text{level}]$  の場合には、指定されたlevelの $a(\text{level})$ 番目に登録されているセル $\text{libs}[\text{level}][a(\text{level})]$ をセルAとし、続いて $b(\text{level}) = a(\text{level}) + 1$ とし、  
 続いて、指定されたlevelに登録されているセル数 $N\text{libs}[\text{level}]$ が、 $b(\text{level}) \leq N\text{libs}[\text{level}]$ か否かを判断する第2判断を行い、  
 前記第2判断の結果、 $b(\text{level}) > N\text{libs}[\text{level}]$  の場合には、 $a(\text{level}) = a(\text{level}) + 1$ とし、前記第1判断を繰り返し行い、  
 前記第2判断の結果、 $b(\text{level}) \leq N\text{libs}[\text{level}]$  の場合には、指定されたlevelの $b(\text{level})$ 番目に登録されているセル $\text{libs}[\text{level}][b(\text{level})]$ をセルBとし、  
 続いて前記セルAと前記セルBとに共通するセルCを構成し、該セルCが空であるか否かを判断する第3判断を行い、  
 前記第3判断の結果、セルCが空である場合には、 $b(\text{level}) = b(\text{level}) + 1$ として、前記第2判断を繰り返し行い、  
 前記第3判断の結果、セルCが空でない場合には、すでに登録されているセルにセルCと同一パターンのセットを持つセルが存在するか否かを判断する第4判断を行い、  
 前記第4判断の結果、セルCと同一パターンのセットを持つセルが存在する場合には、 $b(\text{level}) = b(\text{level}) + 1$ として、前記第2判断を繰り返し行い、  
 前記第4判断の結果、セルCと同一パターンのセットを持つセルが存在しない場合には、当該レベルよりも一つ上のレベルに $N\text{libs}[\text{level} + 1] + 1$ 番目のセルを増設し、 $N\text{libs}[\text{level} + 1] = N\text{libs}[\text{level} + 1] + 1$ とし、該増設したセル $\text{libs}[\text{level} + 1][N\text{libs}[\text{level} + 1]]$ に前記セルCを割り当てて登録し、  
 続いて $\text{level} = \text{level} + 1$ として、一つ上のlevelに対して $\text{TREE}[\text{level}]$ を再帰的に実行するため、前記 $a(\text{level}) = 1$ に戻り、  
 前記第1判断の結果、 $a(\text{level}) > N\text{libs}[\text{level}]$  の場合には、 $\text{level} \leq 1$ か否かを判断する第5判断を行い、  
 前記第5判断の結果、 $\text{level} > 1$ の場合には、 $\text{level} = \text{level} - 1$ として、一つ前のlevelの $\text{TREE}[\text{level}]$ に戻り、 $b(\text{level}) = b(\text{level}) + 1$ として、  
 前記第2判断を繰り返し行い、  
 前記第5判断の結果、 $\text{level} \leq 1$ の場合には、終了することを特徴とするマスクパターンのデータ処理方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、データ処理方法およびマスクパターンのデータ処理方法に関するものであ

6

る。

【0002】

【従来の技術】近年のマスクデータ処理の発展により、データボリュームが巨大化する傾向がある。伝統的なチップ設計では、高密度化と複雑化とによって、既に巨大なマスクデータを出力するようになっている。例えば、 $0.35\mu\text{m}$ 未満のデザインルールにおいては、光近接効果補正、位相シフト技術等のパターン変形をともなうプロセス補正技術等により、データの大きさはさらに増大する可能性が高い。このような状況から、現実的にマスク製作工程における制約（最大ファイルサイズなど）を越える可能性が非常に高くなってきているために、効率の良いマスクデータの圧縮が要求されている。

【0003】

【発明が解決しようとする課題】しかしながら、従来のLSIのマスクパターンの圧縮機能は、マスクパターン描画装置のマスクデータ変換システムに含まれているため、データの圧縮を独立して実行することは困難であった。その上、従来のシステムにはパターンのサイズのような幾何学的情報ではない、例えば露光量を設定するような非幾何学的情報を取り込む機能は無かった。そのため、露光量を補正した後のデータを入力して圧縮することはできなかった。そこで、幾何学的情報に非幾何学的情報を付加してデータを取り扱うデータ処理においても、データ圧縮が必要となっていた。

【0004】本発明は、幾何学的情報とともに非幾何学的情報も取り扱えるようにするとともに、データを効率的に圧縮するのに優れたデータ処理方法およびマスクパターンのデータ処理方法を提供することを目的とする。

【0005】

【課題を解決するための手段】本発明は、上記目的を達成するためになされたデータ処理方法およびマスクパターンのデータ処理方法である。

【0006】すなわち、第1のデータ処理方法は、認識された多数の情報のうちの特定の複数の情報を一つの情報として認識してインデックスを与え、そのインデックスとともに残りの情報とを用いて、データ処理を行う。その際、多数の情報は、幾何学的情報、例えば図形の形状情報やサイズ情報と、非幾何学的情報、例えば露光量情報とからなり、上記非幾何学的情報を仮想的な幾何学的情報として取り扱う。

【0007】上記データ処理方法では、特定の複数の情報を一つの情報として認識してインデックスを与えることから、データが圧縮される。そして、そのインデックスとともに残りの情報を用いて、データ処理を行うことから、データ処理に必要なメモリ数が削減される。また、例えば露光量のような非幾何学的情報を仮想的な幾何学的情報として取り扱うことから、従来は、例えば図形の形状情報やサイズ情報のような幾何学的情報とともにには取り扱いができなかった非幾何学的情報を幾何学的

情報とともにデータ処理することが可能になる。

【0008】第2のデータ処理方法は、複数または単数のパターンからなるものでかつパターン配置が同一であるセルが繰り返し出現する領域を認識する処理を備えたデータ処理方法である。すなわち、 $x-y$ 座標系において、 $x$ 軸方向および $y$ 軸方向に繰り返す状態に配置された複数のセルのうちの一つのセルをこのセルを含む $y$ 軸方向のセル群の中心に位置させることと、上記セルをこのセルを含む $x$ 軸方向のセル群の中心に位置させることを順に繰り返して行うことで、上記セルを $x$ 軸方向および $y$ 軸方向に繰り返す状態に配置された複数のセル領域の中心に位置させる。

【0009】上記第2のデータ処理方法では、一つのセルを、このセルを含む $y$ 軸方向のセル群の中心に位置させることと、上記セルを、このセルを含む $x$ 軸方向のセル群の中心に位置させることを順に繰り返すことから、大きい繰り返し領域に対して、任意の位置をその領域の認識の開始点としても、上記領域を分割することなく、その領域はより大きい一つの繰り返し領域として認識される。そのため、従来は複数の領域群に分割されていた繰り返し領域をより大きく少ない繰り返し領域とすることが可能となるので、データ数は削減される。

【0010】第3のデータ処理方法は、認識された複数のセルからなるセル群を第1レベルとし、各セルに同じパターン集合が存在することを基本にして、この第1レベルのセルからその一部のパターンを受け継いで第1レベルよりも上の第2レベルに新しいセルを構成するセル継承操作を、上記第1レベルのセルおよび新しく構成したセルに対して順次行って、第1レベルのセルからの継承関係を有するセルをツリー状に構成する。そして、各セルについてデータを圧縮するための効果を計算し、最もデータを圧縮するのに効果的なセルを選択して出力する。さらに、上記セル継承操作によって、残ったセルから継承関係を有する新しいセルをツリー状に再構成する操作を、第1レベルのセルが無くなるまであるいはセル継承操作によって継承するセルが見つからなくなるまで繰り返す。

【0011】上記第3のデータ処理方法では、第1レベルのセルからその一部のパターンを受け継いで第1レベルよりも上の第2レベルに新しいセルを構成するセル継承操作を、第1レベルのセルおよび新しく構成したセルに対して順次行うことから、各セルからの継承関係、すなわち、第1レベルのセルおよび新しく構成したセルと共通するパターンを有するセルがツリー状に構成されることになる。そして、これら認識したセルおよび継承により導出されたセルの中より、最も効率のよいセルから順番に利用することで、最適化されたセル表現が実現される。したがって、複数のセルからなるデータは圧縮される。

【0012】マスクパターンのデータ処理方法は、上記

第1～第3のデータ処理方法を用いてマスクパターンのデータ処理を行うものである。

【0013】上記マスクパターンのデータ処理方法では、上記データ処理方法をマスクパターンのデータ処理方法に適用したことから、マスクパターンのデータが圧縮され、従来よりも少ないメモリ数で、従来では処理することができないような巨大なマスクパターンのデータが処理される。そのため、高精度かつ高密度なマスクパターンの製作が可能となる。

【0014】

【発明の実施の形態】本発明の第1実施形態の一例を図1によって以下に説明する。

【0015】図1に示すように、認識された多数の情報として、 $Ia(1)$ 、 $Ia(2)$ 、 $Ia(3)$ 、 $Ib(1)$ 、 $Ib(2)$ 、 $Ib(3)$ 、 $Ic(1)$ 、 $Ic(2)$ 、 $Ic(3)$ という情報があるとする。上記各情報のうちの特定の複数の情報を一つの情報として認識してインデックスを与える。例えば、情報 $Ia(1)$ 、 $Ia(2)$ 、 $Ia(3)$ を一つの情報として、インデックス1を作成する。そして、インデックス1を参照情報として残りの情報 $Ib(1) \sim Ib(3)$ 、 $Ic(1) \sim Ic(3)$ とともに、データ処理する。このように、複数の特定の情報を一つの情報に圧縮することによって、データ処理に必要なメモリ数の削減が可能となる。そのため、巨大データの処理が可能となる。なお、インデックスの作成は、上記組み合わせに限定されることはない。したがって、例えば、 $Ia(1)$ と $Ib(2)$ とでインデックスを作成してもよく、他の情報の組み合わせでインデックスを作成してもよい。

【0016】また、上記各情報を、例えばマスクパターンを構成する各セル（パターン情報）として、マスクパターンを構成する複数のセルのうち、各セルにおいてパターン配置が同一のものでインデックスを作成することも可能である。さらに上記各データを、例えばマスクパターンを構成する各セルとして、そのセルが繰り返し出現する領域を一つのセルとして認識してインデックスを作成することも可能である。すなわち、マスクパターンに対してインデックスを適用することが可能である。

【0017】ここでセル表現できるパターンの繰り返しを、図2によって説明する。図2に示すように、例えばセルの集合を同じ3つのパターン（例えば $\bigcirc \square \triangle$ ）の繰り返しであるとする。ここで、先に示した全てのパターンの定義のかわりに、繰り返されている3つのパターンをもつセルを定義し、セルの繰り返し参照情報を作成する。この例での繰り返しパラメータは繰り返しピッチ $(dx, dy)$ 、繰り返し数 $(5, 2)$ となる。

【0018】このように、多くのパターンデータフォーマットは、単数あるいは複数のパターンをまとめてセルとして構成し、同一パターン群の $x$ 軸、 $y$ 軸に沿った繰り返しを、このセルの繰り返し参照情報として表現する

ことが可能となる。この方法による一つのセル参照情報は、x 軸方向、y 軸方向のそれぞれの繰り返し数、繰り返し間隔あるいは繰り返し領域サイズで表現される。これにより、同じ繰り返しをもつパターンの定義を圧縮することができる。したがって、圧縮されたマスクパターンデータを作成することができる。

【0019】次にパターンデータハッシングについて説明する。まず、配列と比較することによりハッシングテーブルを説明する。例えば、プログラミング言語の一つである C 言語では、以下のような基本的配列が可能である。

a [0], a [1], a [2], ..., a [n-1]  
しかしこれには、2つの大きい制約がある。

(1) アレイサイズが固定であること (配列サイズ n は、実行前に定義しなければならない)。

(2) アレイインデックスは整数のみである (C 言語における [] 内の数値)。

【0020】一方、マスクデータ処理においては、パターンデータ補正等のために可変長の配列が必要となる。また、より複雑なデータ構造をサポートできる配列インデックスも必要となる。これらの課題を解決するためにハッシュテーブルを利用する。このハッシュテーブルでは、例えば、矩形集合の場合、

a [形状情報 1], a [形状情報 2], a [形状情報 3], ...

のように、取り扱う図形の形状情報 (幅、高さなどの複雑な情報) をインデックスとして、目的の図形を取り出すことができる。このハッシュテーブルは、インデックスが複雑なデータ構造であることを除いて、配列のように動作する。

【0021】例えば、特定の矩形を示すデータ構造は、矩形のサイズにより、[幅, 高さ]であり、さらに、描画時にパターン毎に露光量を指定できる EB 描画装置のための入力フォーマットの場合は、仮想的な幾何学情報 (非幾何学的情報) として描画露光量を追加して、従来は個々に取り扱っていた [幅, 高さ, 露光量] に対してインデックスを与えることができる。このようにハッシュ技術により、無理なくデータの拡張ができる。すなわち、インデックスの与え方に制約がない。

【0022】また、パターンデータハッシングを使用することで、計算機メモリの節約が可能となる。例えば、幅 = 2  $\mu$ m、高さ = 3  $\mu$ m の矩形を番号 1、幅 = 2  $\mu$ m、高さ = 4  $\mu$ m の矩形を番号 2 とするように、それぞれのパターンの形状に対して番号を割り当てておくことで、形状情報をこの番号で参照することができる。同一形状のパターンは同じ番号になるため、例えば番号 1 に割り当てた矩形の配置は、間接的参照により、図形の番号とオフセット量のみで表現することができる。このオフセット量は図形の位置を示すことになる。

【0023】直接的な表現である x 座標、y 座標、幅、

高さとを比較すると、直接的な表現では、

x 1 = . . ;

y 1 = . . ;

x 2 = . . ;

y 2 = . . ;

となるが、間接的な表現では、

x = . . ;

y = . . ;

i d = 1 ;

10 [\*\*\*]

となる。したがって、直接的な表現であった幅と高さインデックス i d を作成することによって 1 つの情報になるため、1 つ分の情報が省略できる。

【0024】また、パターンデータハッシングによる他の利点としては、高速なパターン認識ができることがあげられる。与えられたインデックスにより、ハッシュテーブルから、非常に速く関連データを取り出すことができる。さらに、セルなどのパターン集合のように複雑なデータ構造でさえも、パターン集合を、仮想的な幾何学的情報として追加することにより、ハッシュテーブルで取り扱うことができ、それによってセルマッチングが可能になる。

【0025】次にセル認識について説明する。マスクデータフォーマットは一般的に小領域に分割されており、セルの繰り返しは、この領域の中に完全に入っている。したがって、領域毎に独立してセルや繰り返しを認識することができる。それぞれの領域について、パターンデータは計算機メモリ上のハッシュテーブルに入力され、同じパターンの中から繰り返しを認識する。

30 【0026】セルの繰り返しを認識するステップとしては、図 3 の (1) に示すように、x 軸方向、y 軸方向の最大の一次的に繰り返される同一パターン (0 印で示す、\* 印は検索中のセルを示す) を検索する。次に図 3 の (2) に示すように、最大の二次的に繰り返される同一パターン (0 印で示す、\* 印は検索中のセルを示す) を検索する。さらに図 3 の (3) に示すように、最大の同一パターンの繰り返し領域を選択する操作を行い、繰り返しがなくなった時点で検索を終了する。認識された繰り返し領域は 0 印で示す。認識された繰り返し領域のパラメータは保存され、そのパラメータに該当する実際のパターンはハッシュテーブルから削除される。

【0027】次に、同じ繰り返し領域に所属するパターンを集めてセルを作成する。同じセルは任意の小領域から参照することができる。そのため、セルの重複定義を避けるため、セルハッシングによるマッチングを行う。そして最後の領域を処理した後、完成したセルとセル参照情報とが出力可能となる。

【0028】例えば、図 4 に示すフローチャートはセル認識のシステムを示すものである。このシステムは 2 回入力データを読み込む。最初の読み込みでは、セルとそ



の繰り返しと認識する。例えば電子線描画の場合では、単位領域データの入力ですで一つのフィールド領域を認識させ、続いてフィールド領域の全領域を検索して認識させる。そしてフィールド領域のセルを作成する。続いてセルの出力で、フォーマットの範囲内でデータの圧縮効率のよいセルを出力する。そして2回目の読み込みで、各フィールド領域に対して、セルの繰り返し参照と非圧縮パターンを出力する。この2回の読み込み処理は、出力すべき巨大な非圧縮データを、メモリに格納しないほうが好ましいためである。

【0029】次に本発明の第2実施形態の一例を図5の繰り返しセンタトラッキングの説明図によって説明する。

【0030】図5に示すように、この繰り返しセンタトラッキングは、繰り返し認識開始点となるセルを、x軸方向、y軸方向のより大きい繰り返しセル群の中心になるまで移動する操作である。すなわち、x軸方向およびy軸方向に連続的に配置された繰り返しセルF Sのうちの繰り返し認識開始点となるセルFを、(1)上記セルFを含むy軸方向のセル群F g yの中心に位置させる第1操作、(2)上記セルFを含むx軸方向のセル群F g xの中心に位置させる第2操作、とを順に繰り返し行うことで、上記繰り返し認識開始点となるセルFをx軸方向およびy軸方向に連続的にセルが配置されたセルの繰り返し領域F Sの中心に位置させるデータ処理方法である。

【0031】上記繰り返しセンタトラッキングでは、一つのセルFを、それを含むy軸方向のセル群F g yの中心に位置させることと、またセルFを、それを含むx軸方向のセル群F g xの中心に位置させることを順に繰り返すことから、セルの繰り返し領域F Sに対して、任意の位置をその領域の認識の開始点としても、上記領域を分割することなく、その領域は一つの繰り返し領域として認識される。そのため、従来は複数の領域群に分割されていた繰り返し領域をより大きくより少ない繰り返し領域とすることが可能となるので、データ数は削減される。

【0032】次に、上記繰り返しセンタトラッキングを図6のフローチャートおよび前記図5によって具体的に説明する。

【0033】図6に示すように、「START」でフローを開始する。まず、「 $n=N$ 」で、最大繰り返し数をNとして、繰り返し数 $n=N$ に設定する。次いで「 $n>0?$ 」によって、繰り返し数が上限になっているか否かを判断する第1判断を行う。

【0034】上記第1判断の結果、「YES」、すなわち $n>0$ の場合には、x軸方向およびy軸方向に連続的に配置された繰り返しセルF S（以下、セルF Sに関しては図5参照）のうちの一つのセルF（以下、セルFに関しては図5参照）に着目する。そして、上記セルFが

このセルFを含むy軸方向のセル群F g y（以下、セルF g yに関しては図5参照）を、「y軸方向の繰り返しセル群F g yを得る」によって求める。その後、「セルFの位置≠セル群F g yの中心?」によって、上記セルFが上記セル群F g yの中心に位置していないか否かを判断する第2判断を行う。

【0035】前記第2判断の結果、「YES」、すなわちセルFがそれを含むy軸方向のセル群F g yの中心に位置しない場合には、「セルFの位置をセル群F g yの中心に移動」によって、セルFを上記y軸方向のセル群F g yの中心に移動する。そして、「 $n=n-1$ 」によって、繰り返し数を1減じて、上記第2判断を繰り返し行う。

【0036】前記第2判断の結果、「NO」、すなわちセルFがそれを含むy軸方向のセル群F g yの中心に位置する場合には、セルFがこのセルFを含むx軸方向のセル群F g x（以下、セルF g xに関しては図5参照）を、「x軸方向の繰り返しセル群F g xを得る」によって求める。そして「セルFの位置≠セル群F g xの中心?」によって、上記セルFが上記セル群F g xの中心に位置していないか否かを判断する第3判断を行う。

【0037】上記第3判断の結果、「YES」、すなわちセルFがそれを含むx軸方向のセル群F g xの中心に位置していない場合には、「セルFの位置をセル群F g xの中心に移動」によって、セルFを上記x軸方向のセル群F g xの中心に移動する。そして、「 $n=n-1$ 」によって、繰り返し数を1減じて、上記第2判断を繰り返し行う。

【0038】前記第3判断の結果、「NO」、すなわちセルFがそれを含むx軸方向のセル群F g xの中心に位置する場合には、上記セルFの位置をx軸方向およびy軸方向に連続的に配置された繰り返しセルの中心として終了する。また、上記第1判断の結果、「NO」、すなわち $n \leq 0$ ならば終了する。

【0039】なお、上記フローチャートでは、理論的な無限トラッキングを避けるために最大繰り返し数Nを設定した。もし、最大繰り返し数Nを設定しない場合には、例えば、図7に示すような繰り返しセルの場合には無限トラッキングを行うことになる。

【0040】すなわち、y軸方向のセル群F g y 1の中心のセルF y 1に対して、x軸方向のセル群F g x 1の中心のセルは、セルF x 1になる。このセルF x 1に対して、y軸方向のセル群F g y 2の中心のセルは、セルF y 2になる。そしてセルF y 2に対して、x軸方向のセル群F g x 2の中心のセルは、セルF x 2になる。さらにセルF x 2に対して、y軸方向のセル群F g y 3（＝セル群F g y 1）の中心のセルは、セルF y 1になる。このように繰り返すことによって、初めのセルF y 1の位置に戻るため、上記のように最大繰り返し数Nを設定する必要がある。

【0041】上記繰り返しセンタトラッキングによれば、より大きな繰り返し領域を分割することがなくなる。すなわち、同一データの繰り返しを認識する際、任意の場所を認識の開始点とすると、より大きな繰り返し領域を分割してしまう危険性がある。例えば図8に、繰り返し開始点により大きい繰り返し領域を分割してしまう例を示す。

【0042】図8の(1)に示すように、セルFSにおいて、例えば図面の左側のセルからx軸方向に繰り返しセルを認識する。そして、\*印の位置では、y軸方向の繰り返しはないため、x軸方向のみの繰り返しとして、セル群Fgxを認識して登録することとなる。

【0043】続いてセルFSの他の繰り返しを認識する。図8の(2)に示すように、上記セル群Fgxは登録したので、セルFSは二つのセル群FS1とFS2とに分割された状態になる。このように大きい繰り返し領域となるセルFSを分割してしまうと、より大きく効果的な繰り返しを構成する機会を減らしてしまうことになる。したがって、上記繰り返しセンタトラッキングを用いれば、常に繰り返し領域の中心に向かって、中心の位置を検索するため、上記のような問題は回避される。そのため、より大きなデータを一つのデータと認識することが可能となるので、データの圧縮効率は高くなる。

【0044】上記繰り返しセンタトラッキングは、マスクパターンのデータ処理に適用することが可能である。すなわち、一つのセルFを、このセルFを含むy軸方向のセル群Fgyの中心に位置させることと、セルFを、このセルFを含むx軸方向のセル群Fgxの中心に位置させることとを順に繰り返すことから、マスクパターンの大きな繰り返し領域に対して、任意の位置をその領域の認識の開始点としても、上記領域を分割することなく、その領域は一つの繰り返し領域として認識される。そのため、従来は複数の領域群に分割されていた繰り返し領域をより大きくより少ない繰り返し領域とすることが可能となるので、マスクパターンのデータ数は削減される。

【0045】次に本発明の第3実施形態の一例を、図9のセル継承ツリーの説明図によって説明する。

【0046】図9に示すように、認識された複数のセルを第1レベルとする。そして、例えばA(1), A(2), A(3), A(4)に同じパターン集合が存在することを基本にして、第1レベルのセル、A(1)~A(4)からその一部のパターンを受け継いで第1レベルよりも上の第2レベルに新しいセル、例えばB(1), B(2), B(3)を構成するセル継承操作を、第1レベルのセル、A(1)~A(4)および新しく構成したセル、B(1)~B(3), C(1), C(2), D(1)に対して順次行って、第1レベルのセル、A(1)~A(4)からの継承関係を有するセル、B(1)~B(3), C(1), C(2), D(1)を

ツリー状に構成する。

【0047】続いて、認識されたセル、A(1)~A(4)、および各新しいセル、B(1)~B(3), C(1), C(2), D(1)についてデータを圧縮するための効果を計算し、データを圧縮するのに最も効果的なセルを選択して出力する。その後、上記セル継承操作によって、残ったセルから継承関係を有する新しいセルをツリー状に再構成する操作を、第1レベルのセルが無くなるまであるいは上記セル継承操作によって継承するセルが見つからなくなるまで繰り返す。

【0048】上記セル継承操作では、第1レベルのセルからその一部のパターンを受け継いで第1レベルよりも上の第2レベルに新しいセルを構成するセル継承操作を、第1レベルのセルおよび新しく構成したセルに対して順次行うことから、各セルからの継承関係、すなわち、第1レベルのセルおよび新しく構成したセルと共通するパターンを有するセルがツリー状に構成されることになる。そのため、共通するパターンを有するセルを用いて、第1レベルのセルを表すことによって、第1レベルの各セルを構成する個々のパターンを全て登録する必要がなくなる。したがって、複数のセルからなるデータは圧縮される。

【0049】上記セル継承操作は、マスクパターンのデータ処理に適用することが可能である。すなわち、マスクパターンとして認識された複数のセルに同じパターン集合が存在することを基本にして、マスクパターンのセルからその一部のパターンを受け継いで上位レベルに新しいセルを構成するセル継承操作を、マスクパターンのセルおよび新しく構成したセルに対して順次行って、マスクパターンのセルからの継承関係を有するセルをツリー状に構成する。

【0050】続いて、各新しいセルについてデータを圧縮するための効果を計算して、最もデータを圧縮するのに効果的なセルを選択して出力する。その後、上記セル継承操作によって、残ったセルから継承関係を有する新しいセルをツリー状に再構成する操作を、マスクパターンのセルが無くなるまで、あるいは上記セル継承操作によって継承するセルが見つからなくなるまで繰り返す。

【0051】上記のように、マスクパターンに対してセル継承操作を行うことで、マスクパターンは共通するパターンを有するセルを用いて表すことが可能となる。そのため、マスクパターンを構成する各パターンを個別に登録する必要が無くなるために、マスクパターンのデータは圧縮されることになる。

【0052】次に、上記セル継承ツリーを形成するためのセル継承操作を、図10のフローチャートによって具体的に説明する。ここでは、指定されたlevelにあるデータ処理方法TREE[level]の実行を説明する。

【0053】まず、「a(level)=1」によって、a(level)=1とする。そして「a(level)≤Nlibs

「level」?」によって、指定されたlevelに登録されているセル数Nlibs[level]が $a(\text{level}) \leq Nlibs[\text{level}]$ か否かを判断する第1判断を行う。

【0054】上記第1判断の結果、「YES」、すなわち $a(\text{level}) \leq Nlibs[\text{level}]$ の場合には、「 $A = libs[\text{level}][a(\text{level})]$ 」によって、指定されたlevelの $a(\text{level})$ 番目に登録されているセル $libs[\text{level}][a(\text{level})]$ をセルAとする。続いて、「 $b(\text{level}) = a(\text{level}) + 1$ 」によって、 $b(\text{level})$ を求める。その後、「 $b(\text{level}) \leq Nlibs[\text{level}]$ 」によって、指定されたlevelに登録されているセル数Nlibs[level]が、 $b(\text{level}) \leq Nlibs[\text{level}]$ か否かを判断する第2判断を行う。

【0055】上記第2判断の結果、「NO」、すなわち $b(\text{level}) > Nlibs[\text{level}]$ の場合には、「 $a(\text{level}) = a(\text{level}) + 1$ 」によって、 $a(\text{level})$ に1を加算して新たな $a(\text{level})$ を求めてから、上記第1判断を繰り返す行。

【0056】上記第2判断の結果、「YES」、すなわち $b(\text{level}) \leq Nlibs[\text{level}]$ の場合には、「 $B = libs[\text{level}][b(\text{level})]$ 」によって、指定されたlevelの $b(\text{level})$ 番目に登録されているセル $libs[\text{level}][b(\text{level})]$ をセルBとする。続いて「 $A \cap B = C$ が空?」によって、上記セルAと上記セルBとに共通するセルCを構成し、セルCが空か否か、すなわちセルCが存在しないか否かを判断する第3判断を行う。

【0057】上記第3判断の結果、「YES」、すなわちセルCが空である場合には、「 $b(\text{level}) = b(\text{level}) + 1$ 」によって、 $b(\text{level})$ に1を加算して新たな $b(\text{level})$ を求める。そして上記第2判断を繰り返す行。

【0058】上記第3判断の結果、「NO」、すなわちセルCが空でない場合には、「セルCと同一パターンのセットを持つセルが存在しないか?」既に登録されているセルにセルCと同一パターンのセットを持つセルが存在しないか否かを判断する第4判断を行う。

【0059】上記第4判断の結果、「NO」、すなわちCと同一パターンのセットを持つセルが存在する場合には、「 $b(\text{level}) = b(\text{level}) + 1$ 」によって、 $b(\text{level})$ に1を加算して新たな $b(\text{level})$ を求める。そして、上記第2判断を繰り返す行。

【0060】上記第4判断の結果、「YES」、すなわちCと同一パターンのセットを持つセルが存在しない場合には、「 $Nlibs[\text{level} + 1] = Nlibs[\text{level} + 1] + 1$ 」によって、上記levelよりも一つ上のレベルにセル $libs[\text{level} + 1][Nlibs[\text{level} + 1]]$ を増設する。そして、「 $libs[\text{level} + 1][Nlibs[\text{level} + 1]] = C$ 」によって、増設したセルにCを割り当てて登録する。続いて「level = level

+ 1」によって、上記levelに1を加算して、一つ上のlevelを求める。さらに「TREE[level]」によって、一つ上のlevelに対してTREE[level]を再帰的に実行するため、上記「START」に戻る。

【0061】一方、上記第1判断の結果、「NO」、すなわち $a(\text{level}) > Nlibs[\text{level}]$ の場合には、「level ≤ 1?」によってlevelが1以下か否かを判断する第5判断を行う。

【0062】上記第5判断の結果、「NO」、すなわち $level > 1$ の場合には、「level = level - 1」によって、levelから1を減じて一つ下の新たなlevelを求める。そして、一つ前のlevelの「TREE[level]」に戻り、「 $b(\text{level}) = b(\text{level}) + 1$ 」によって、 $b(\text{level})$ に1を加算して新たな $b(\text{level})$ を求める。そして、上記第2判断を繰り返す行。

【0063】上記第5判断の結果、「YES」、すなわち $level \leq 1$ の場合には、「END」になり終了する。

【0064】次に上記セル継承ツリーを用いてデータの圧縮を行った場合の効果、図11の実験データによって説明する。図では、縦軸にセル展開図形数(Log)を示し、横軸にセルの種類を圧縮効率の高い順に示す。また、0印はセル継承ツリーを用いない場合で、+印はセル継承ツリーを用いてデータを圧縮した場合である。図に示すようにセル継承ツリーを用いて共通セルを見いだすことによって、少ないセルの種類で多くのパターン数が表現される。すなわち、セル継承ツリーを用いることによって、データの圧縮率は向上する。

【0065】なお、上記説明したデータの圧縮においては、圧縮されなかったパターンデータの出力時には、データは既に分類されている。そのため、同じ図形種(矩形、台形等)の図形を連続して出力することができる。さらに、冗長な図形種の再定義を省略ができるマスクパターンのフォーマットであれば、さらにデータを圧縮することができる。

【0066】例えば、図形種を任意の順番に配列して出力した場合に以下のようにする。

```
rectangle #1、
coordinates #1、
x-trapezoid #2、
coordinates #2、
rectangle #3、
coordinates #3、
x-trapezoid #4、
coordinates #4、
rectangle #5、
coordinates #5、
```

【0067】ここで、上記出力した図形種の配列は、同一図形種の連続出力として、以下のようにする。

```
rectangle #1、
coordinates #1、
```

coordinates #3、  
 coordinates #5、  
 x-trapezoid #2、  
 coordinates #2、  
 coordinates #4、  
 [\*\*\*\*\*]、  
 [\*\*\*\*\*]、  
 [\*\*\*\*\*]、  
 すなわち、初めに、「rectangle #1、coordinates #1」が出力されているため、次に出てくる「rectangle #3、coordinates #3」は「coordinates #3」で代表されて「rectangle #3」は省略される。同様に、「coordinates #4」、「coordinates #5」で代表される「x-trapezoid #4」、「rectangle #5」も省略される。そのため、データの圧縮が可能となる。

【0068】上記説明したようなマスクデータの圧縮技術をプログラミング言語Cで作成して、135MIPSのUNIXワークステーションによって、0.25μmルールの中の多くのメモリをもつASICのパターンデータでテストを行った。マスクデータフォーマットには、ここでは、JEOL52フォーマットを用いた。これは、セルをライブラリという名称で実現している。上記条件にて、本発明の方法による露光量情報も考慮に入れたデータ圧縮処理では、885.4Mbytesのデータを7.2Mbytesに3.5時間で圧縮することができた。

【0069】従来のシステム（例えば、並列プロセッサの合計320MIPSのシステム）によるデータ圧縮処理では、単独での圧縮処理ができないため、アウトライン処理、描画データへの分割処理を含め、ほぼ同じ処理時間でファイルサイズは約2倍であった。またこの処理では、露光量情報は取り扱えないため、考慮していない。

【0070】以上、条件が異なるため、単純な比較は困難ではあるが、本発明の方法を用いることによって、高効率かつ高圧縮率のデータ処理が実現できる。一般的に、露光量を設定したデータは展開され、そのままではファイルサイズが描画装置の許容範囲を越えてしまうケースがある。そこで、データを圧縮することにより初めて描画が可能となる。本発明のデータ処理方法によればより大規模なデータも描画することが可能になる。

#### 【0071】

【発明の効果】以上、説明したように本発明によれば、認識された多数の情報のうちの特定の複数の情報を一つの情報として認識してインデックスを与えるので、データを圧縮することが可能になる。また、非幾何学的情報を仮想的な幾何学的情報として取り扱うので、従来は取り扱えなかった露光量情報のような非幾何学的情報も幾何学的情報とともに処理することが可能になる。そのため、従来は大規模に展開する必要があった非幾何学的情

報の規模が小さくなる。したがって、データ処理に必要なメモリ数が削減されるので、より大規模なデータ処理が可能になる。

【0072】また、一つのセルをこのセルを含むy軸方向のセル群の中心に位置させることと、上記セルをこのセルを含むx軸方向のセル群の中心に位置させることとを順に繰り返す、いわゆる繰り返しセンタトラッキングによれば、パターンの大きな繰り返し領域に対して、任意の位置をその領域の認識の開始点としても、上記領域を分割することなく、その領域は一つの繰り返し領域として認識することが可能になる。そのため、従来は複数の領域群に分割されていた繰り返し領域をより大きくより少ない繰り返し領域とすることが可能となるので、データ数を削減することができる。

【0073】さらに、第1レベルのセルからその一部のパターンを受け継いで第1レベルよりも上の第2レベルに新しいセルを構成するセル継承操作を、第1レベルのセルおよび新しく構成したセルに対して順次行うので、各セルからの継承関係、すなわち、第1レベルのセルおよび新しく構成したセルと共通するパターンを有するセルがツリー状に構成できる。そして、これら認識したセルおよび継承により導出されたセルの中より、最も効率の良いセルから順番に利用することで、最適化されたセル表現が実現できる。したがって、複数のセルからなるデータを圧縮することが可能になるので、より大規模なデータ処理が可能になる。

【0074】またさらに、本発明のデータ処理方法をマスクパターンのデータ処理方法に適用したので、マスクパターンのデータが圧縮され、従来よりも少ないメモリ数で、従来では処理することができないような巨大なマスクパターンのデータを処理することが可能になる。

#### 【図面の簡単な説明】

【図1】本発明に係わる第1実施形態のインデックスの説明図である。

【図2】セル表現できるパターンの繰り返しの説明図である。

【図3】セルの繰り返しを認識するステップの説明図である。

【図4】セル認識のシステムのフローチャートである。

【図5】第2実施形態の繰り返しセンタトラッキングの説明図である。

【図6】繰り返しセンタトラッキングのフローチャートである。

【図7】無限トラッキングの説明図である。

【図8】繰り返し認識時の副作用の説明図である。

【図9】第3実施形態のセル継承ツリーの説明図である。

【図10】セル継承操作のフローチャートである。

【図11】データ圧縮の効果の説明図である。

【符号の説明】

## 1 インデックス

A (1) ~ A (4) 第1レベルのセル

B (1) ~ B (3) 新しいセル

C (1), C (3) 新しいセル

D (1) 新しいセル

F セル

F y g y 軸方向のセル群

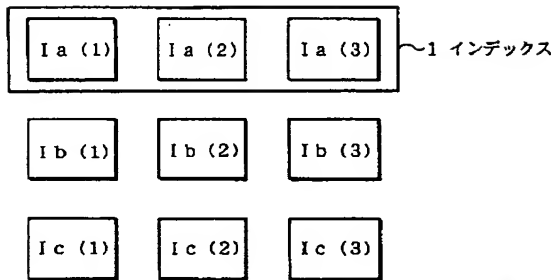
F y x x 軸方向のセル群

I a (1) ~ I a (3) 情報

I b (1) ~ I b (3) 情報

I c (1) ~ I c (3) 情報

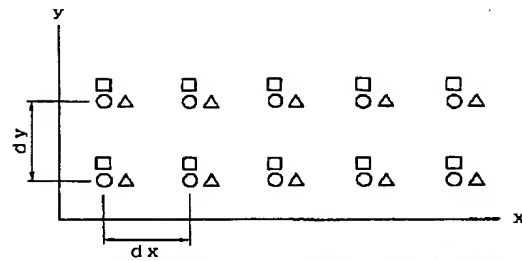
【図1】



I a (1) ~ I a (3), I b (1) ~ I b (3), I c (1) ~ I c (3): 情報

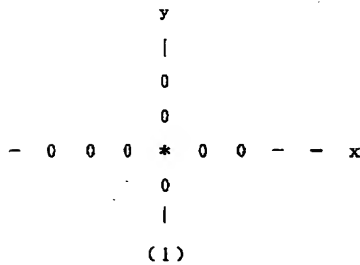
本発明に係わる第1実施形態のインデックスの説明図

【図2】



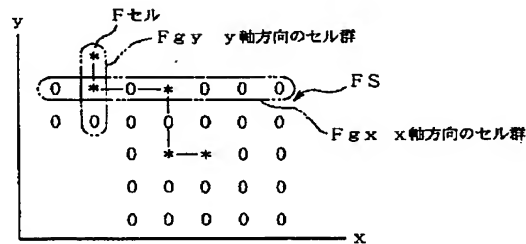
セル表現できるパターンの繰り返しの説明図

【図3】

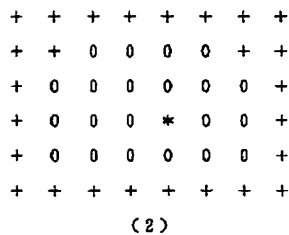


(1)

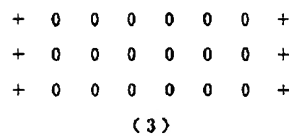
【図5】



第2実施形態の繰り返しセンタトラッキングの説明図

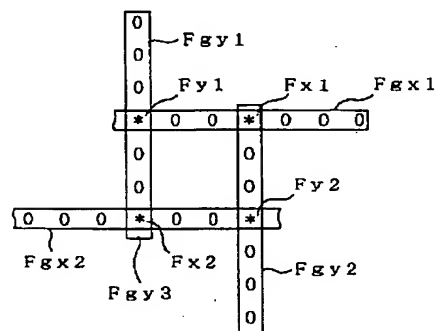


(2)



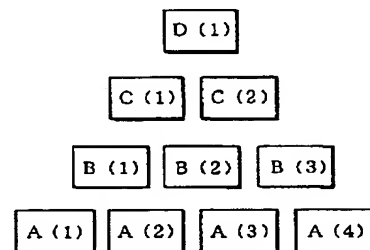
(3)

【図7】



無限トラッキングの説明図

【図9】



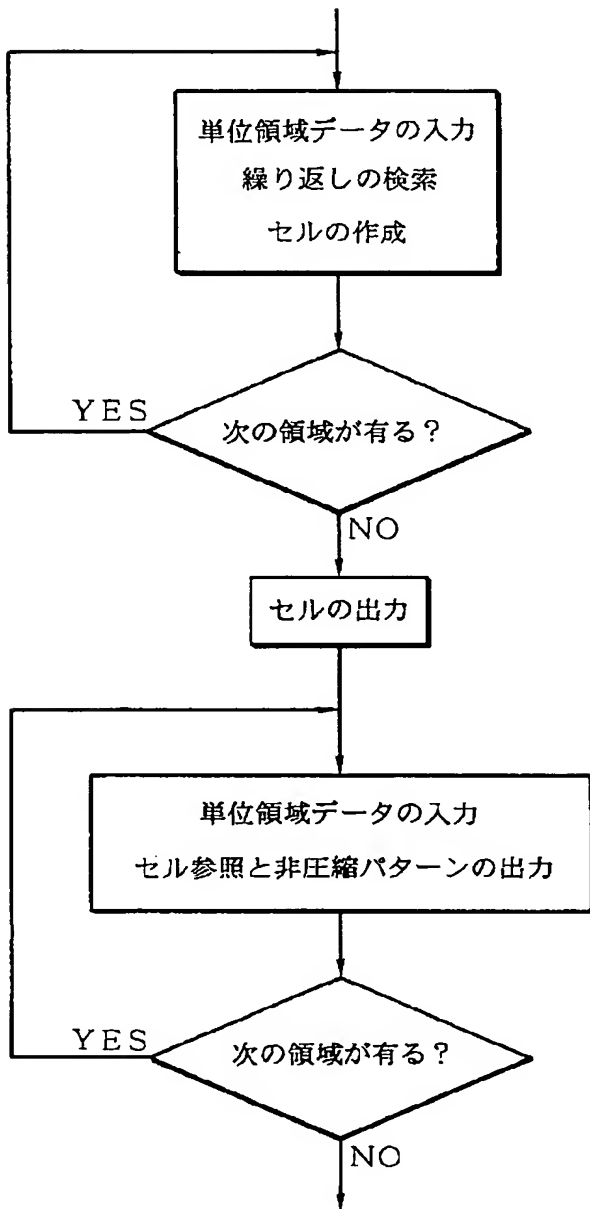
A (1) ~ A (4): 第1レベルのセル

B (1) ~ B (3), C (1), C (2), D (1): 新しいセル

第3実施形態のセル継承ツリーの説明図

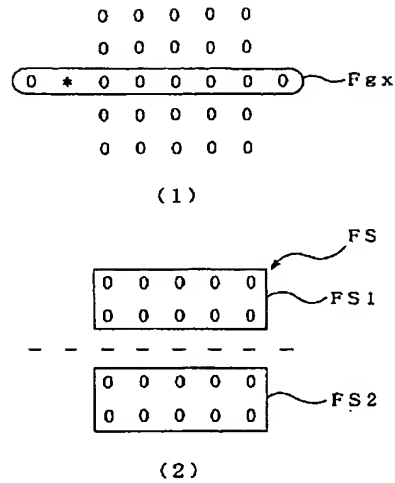
セルの繰り返しを認識するステップの説明図

【図4】



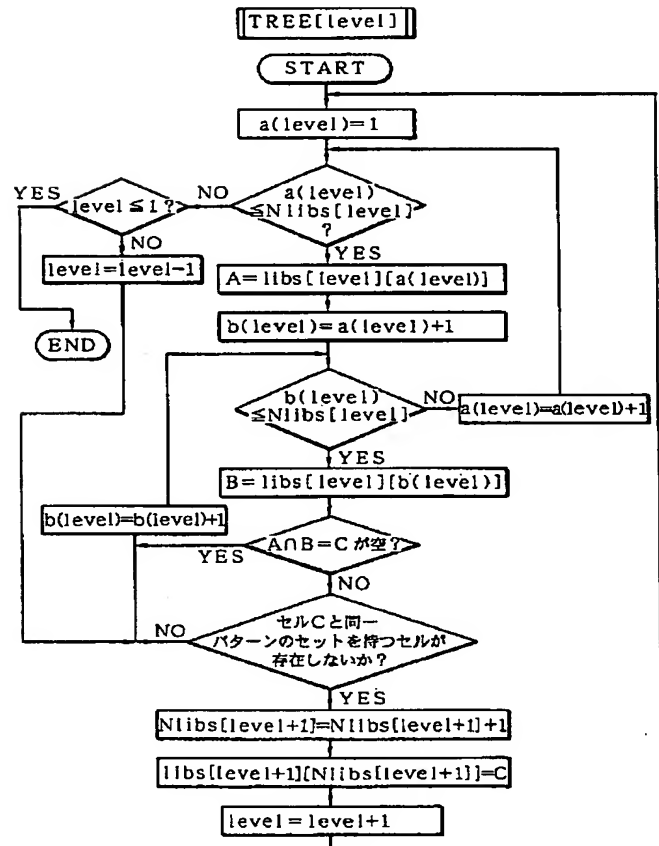
セル認識のシステムのフローチャート

【図8】



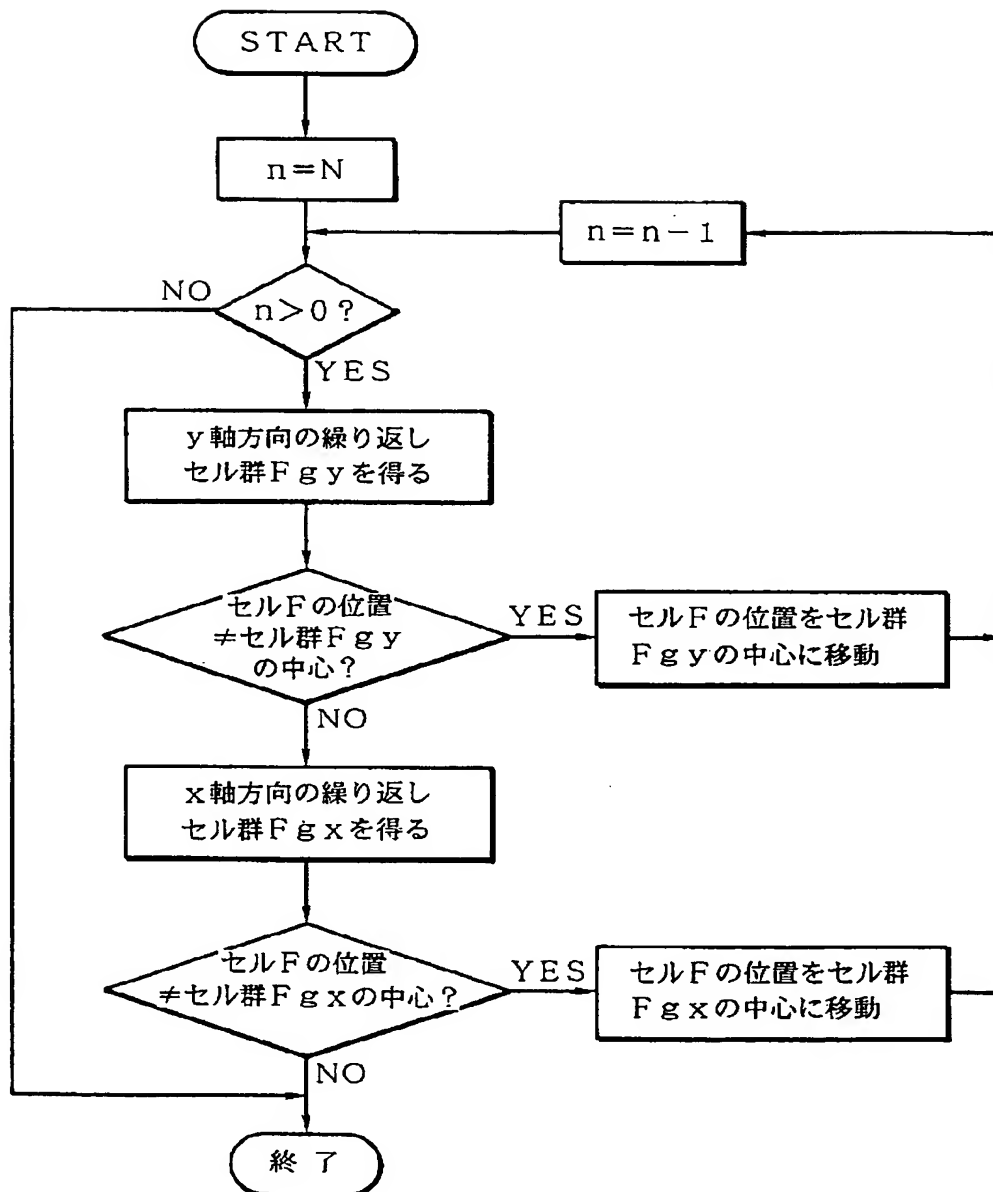
繰り返し認識時の副作用の説明図

【図10】



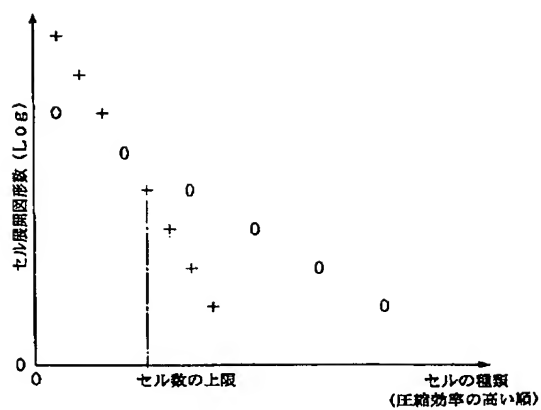
セル継承操作のフローチャート

【図 6】



繰返しセンタトラッキングのフローチャート

【図 11】



データの圧縮効果の説明図